

A Tool For Automatic Verification of Real-Time Expert Systems

Bonnie Traylor

Jet Propulsion Laboratory

California Institute of Technology

4800 Oak Grove Drive

Pasadena, CA 91109

(81 8) 354-1449

bonnie@puente.jpl.nasa.gov

1.0 introduction

The creation of an automated, user-driven tool for expert system development, validation, and verification is currently ongoing at NASA's Jet Propulsion Laboratory. In the new age of "faster, better, cheaper" missions, there is an increased reliance upon embedded expert systems in order to encapsulate and preserve mission expertise. The expert systems currently in use at JPL are responsible for real-time monitoring and diagnosis over broad domains [Schwuttke, et al. 1994]. These systems are responsible for efficient diagnosis of complex system failures in real-time environments with high data volumes and moderate failure rates. In order to ensure high performance, the algorithmic portions of the systems are implemented in C, while the knowledge-based diagnostic modules are called upon primarily for decision-making. The experts involved are the spacecraft analysts, and they are themselves the end-users. Therefore, the primary motivation behind the development of this tool is maximization of expert time and resources by facilitation of development and updates of knowledge bases, and the creation of a standard platform over which expert systems can be built and maintained, eliminating the need for understanding several shell languages. Toward this end, this tool consists of several major components: an update and development tool, utilizing a meta-language and templates for rule definition; a library of standard checkers as commonly

discussed in the literature, including a performance monitor; and a visual guide to the rules in the form of a flow graph displaying the relationships among rules, all packaged in a sophisticated graphical user interface.

2.0 Development/Update Component

Crossing the language barrier is a major feat in the development of an expert system verification tool in an industry where shell languages are plentiful, and standardization is virtually impossible. An analysis of rulebases currently in use for monitoring downlink telemetry, however, has revealed that the vast majority of rules utilize only a fraction of the semantic components of the language, and these components seem to vary only slightly from shell to shell. In addition, it would seem that allowing the experts to encode knowledge themselves, with only a little outside assistance, would greatly facilitate the ES development process, getting expertise on-line quicker, and with fewer errors due to miscommunication between experts and knowledge engineers. However, many shells utilize such complicated syntax and elaborate functionality that experts are understandably hesitant to undertake the task of learning the language in addition to their other responsibilities. In order to overcome these difficulties, it seems logical to create a meta-language with very simple syntax and few keywords which embodies the commonalities of ES shell semantics, without the overhead of bulky syntax and functionality. The structure of the rules can be presented to the user in the form of

The research described in this abstract was carried out by the Jet Propulsion Laboratory, California Institute of Technology under a contract with the National Aeronautics and Space Administration.

templates and pushbuttons. Then, as in EVA [Stachowitz et. al., 1987], COVER [Preece, et. al. 1992a], and others, the validation and verification tests can be performed on the meta-language rules. These verified rules can then be translated into the shell language of choice with minimal effort.

3.0 verification Component

Various issues were considered in the determination of knowledge base tests to include in this tool. Because simplicity and ease of use are primary concerns, the user will enter candidate rules into the rulebase via a template, which will provide the user with the appropriate format of the rule. Before a candidate rule is accepted for insertion into the rulebase, the rule will undergo various tests for conditions such as inconsistency, subsumption (redundancy), and looping, among others, ([Preece et al. 1992b] provides concise, clear definitions of verification tests performed by several existing systems, in addition to providing a framework for comparison.) Syntactic checking, such as that used by CHECK [Nguyen et al. 1985] will provide a core, however, initial analysis indicates that extended semantic checking like that described in [Suwa, et al., 1982] and [Stachowitz, et al., 1987] will also be possible, because the domains are mutually exclusive. The tool will also include capabilities for dynamic analysis of the knowledge base -- primarily for determination of frequently traversed paths, frequently and infrequently used rules, and other performance meters. A display of graphical elements corresponding to the rules and the relationships among them will also be included in the tool. This is clearly a very effective medium for the presentation of certain types of errors and also for performance monitoring. Although neither the graphical display of the rules, nor the graphical user interface in which the tool will be packaged is of any interest in ES V&V research, they are essential components of a real-world application -- a tool which is too abstruse or esoteric to be used with minimal effort on the part of the expert/user is of little value for producing usable systems.

4.0 Conclusion

Utilization of a meta-language front-end to multiple expert system shells, combined with powerful verification algorithms and a sophisticated graphical user interface provides a practical solution to an old problem. With the use of expert systems on the rise, the development of such tools is critical.

5.0 References

- Nguyen, T. A.; Perkins, W. A.; Laffey, T. J.; and Pecora, D., 1985. "Checking an Expert Systems Knowledge Base for Consistency and Completeness." In *Proc. 9th Internat 'l Joint Conference on Artificial intelligence*, Vol 1, 375-378, AAAI.
- Preece, A. D.; Shinghal, R.; and Batarehk, A., 1992. "Principles and Practice in Verifying Rule-Based Systems," In *The Knowledge Engineering Review*, Vol 7:2, 115-141.
- Preece, A. D.; Shinghal, R.; and Batarehk, A., 1992. "Verifying Expert Systems: a Logical Framework and a Practical Tool." *Expert Systems with Applications* 4(2/3).
- Schwuttke, U, M.; Veregge, J. R.; and Quan, A. G., 1994. "Cooperating Expert Systems for the Next Generation of Real-time Monitoring Applications," In *Proc. 2nd Internat'l Conference on Expert Systems for Development*, Asian Institute of Technology, Bangkok, Thailand.
- Stachowitz, R. A.; and Combs, J. B., 1987. "Validation of Expert Systems." In *Proc. 20th Annual Hawaii International Conference on System Sciences*, Vol 1, 686-695.
- Suwa, M.; Scott, A. C.; and Shortliffe, E. H., 1982. "An Approach To Verifying Completeness and Consistency in a Rule-based Expert System." *AI Magazine*, 3(4) 16-21.